LISBON
SCHOOL OF
ECONOMICS &
MANAGEMENT
UNIVERSIDADE DE LISBOA

MASTER IN

APPLIED ECONOMETRICS AND FORECASTING

MASTER'S FINAL WORK

DISSERTATION

INTERPRETABLE MODELS OF LOSS GIVEN DEFAULT

SARA MADEIRA MATOS

JANUARY - 2021

LISBON
SCHOOL OF
ECONOMICS &
MANAGEMENT
UNIVERSIDADE DE LISBOA

MASTER IN

APPLIED ECONOMETRICS AND FORECASTING

MASTER'S FINAL WORK

DISSERTATION

INTERPRETABLE MODELS OF LOSS GIVEN DEFAULT

SARA MADEIRA MATOS

ADVISOR:

JOÃO A. BASTOS

JANUARY - 2021

ABSTRACT

Model interpretability may be defined as the degree to which a human can understand the cause of its outputs. In statistical modeling there is a trade-off between interpretability and prediction accuracy: interpretable models are usually less accurate, whereas complex models have greater out-of-sample precision so long as overfitting is controled. Credit risk management is an area where regulators expect financial institutions to have transparent and auditable risk models, which currently leads to a constraint in the use of black-box models. Furthermore, unknown biases in risk models may lead to unfair lending decisions.

This work aims to investigate whether a black-box model of Machine Learning (Extreme Gradient Boosting (XGBoost)) can outperform the Natural Interpretable Models (Fractional Response Model (FRM) and Regression Tree (RT)) in the Loss Given Default (LGD) prediction and evaluate whether financial institutions have a chance of not sacrificing the transparency of the model for the quality of precision by using black-box models. If financial institutions are able to correctly predict and interpret the risk drivers of LGD, it can lead to effective gains in the calculation of both their regulatory capital requirements and the price of its financial products, which may generate a competitive advantage in the financial market.

For the analysis of the degree of interpretability of XGBoost, this work uses the Shapley Additive exPlanations (SHAP) method, the Permutation Feature Importance method, and the Partial Dependence Plots. We show that XGBoost predicts LGD better than the Natural Interpretable Models considered and that its outputs can be interpreted without much effort in terms of their inputs. Therefore, banks are able to guarantee the interpretation of their models while pursuing a competitive advantage.

KEYWORDS: Loss Given Default; XGBoost; Model interpretability; Black-Box model.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my advisor Professor João Bastos for his significant contributions and support throughout this journey.

I would also like to thank my family, especially my parents and my twin for their decisive supportive actions.

I am also grateful to Francisco Mendonça, Carolina Vasconcelos and Cinthia Castilho for their comments and long discussions that contributed to the present work.

Finally, a very special thanks to Nuno Aparício for his unconditional support.

1. INTRODUCTION

Credit is generally defined as an agreement in which the debtor receives something of value now and agrees to repay it to creditor later. The availability of credit is one of the factors capable of boosting an economy, as it enables people to consume and accelerates the development of companies. On the other hand, poor management of credit granting by the financial market can potentially lead to financial risks with disastrous consequences for the economy. A recent example of this is the 2008 financial crisis, which was based on the granting of real estate financing to high-risk clients in the USA and which led to a financial crisis that has been felt worldwide for several years. Crises of this kind result in high costs for taxpayers and are a factor of market instability in the economy, leading to acute losses of products and sharp increases in unemployment. Thus, capital requirements are extremely important for financial stability since, at a reasonable cost, they reduce the possibility of bankruptcy in the banking system.

The Basel Agreement of 1988 (BCBS, 1988) marked the beginning of the convergence of the various approaches adopted by different countries for the definition of minimum capital requirements. In June 2004, a review of this regulatory framework, commonly referred to as Basel II, was published by the Basel Committee on Banking Supervision (2004).

Basel II is based on three pillars that reinforce each other. Pillar I includes capital requirements for credit risk, market risk, and operational risk. With this review, capital requirements have come to depend on the quality of credit inferred from estimates of risk factors such as the probability of default (PD), the amount due at the time of default (EAD - Exposure at Default), and the loss given default (LGD). Pillar II concerns banking supervision, with bank supervisors having greater authority to assess the consistency and soundness of risk assessment methodologies developed by banks. Finally, Pillar III introduces rules on the information that banks are required to publish (usually referred to as market discipline).

This thesis contributes to the expanding strand of literature studying the Loss Given Default. The LGD risk factor represents the exposure percentage of an operation that the Bank estimates to lose if the borrower of that operation is no longer able to comply with the contractual terms. This parameter directly impacts the amount of capital that the financial institution must hold in reserves, as required by financial regulators. Thus, accurate estimates of potential losses are essential for an efficient allocation of regulatory and economic capital. Therefore, banks can gain a competitive advantage by improving their LGD projections, since it optimizes minimum capital requirements.

The modeling and forecasting of the Loss Given Default or of its complement, the recovery rate (RR), has been done in many different manners. The first works on this topic used mainly parametric models such as the linear least squares regression (Acharya et al., 2007; Caselli et al., 2008; Davydenko and Franks, 2008 and Grunert and Weber, 2009), the fractional response model (Dermine and Neto de Carvalho, 2006) and the beta distribution (Gupton and Stein, 2005). More recently, non-parametric models have become more frequent, namely through machine learning models (Bastos, 2010; Qi and Zhao, 2011; Loterman et al., 2012; Bastos, 2014; Yao et al., 2015).

Machine learning models are statistical algorithms that minimize a particular cost or loss function by delving into the data's relationships. Some of the work developed prove that machine learning models (for example, neural networks) outperform the simple linear models and the decision trees in predictive performance since they can model complex relationships in the data. However, credit risk management is an area where regulators expect financial institutions to adopt transparent and auditable risk models. The opacity of these models, since they do not provide a direct explanation for their forecasts, makes financial institutions reluctant to use them. This leads to the dilemma between precision and interpretability of a model, because normally, the more interpretable a model is, the less accurate it is. This is because natural interpretable models are simple and simple models do not have the flexibility to capture complex ideas. In addition, uncertainty in the use of the interpretation models developed makes financial institutions afraid to make unfair decisions when granting credit due to unknown biases in risk models that may result from an inaccurate analysis of black-box models. Another difficulty felt by financial institutions is the learning cost associated with black-box models that must be overcome by risk analysts. This makes natural interpretable models more attractive, since these models readily attribute the relationship between LGD and regressors.

Recognizing the enormous predictive capacity of machine learning models, the debate has emerged around techniques for making machine learning models more interpretable and overcoming the difficulties identified in the use of black-box models. Some of these techniques are: the Global surrogate method (Craven and Shavlik, 1996), the Permutation Feature Importance method (Fisher et al., 2018), the Partial Dependence Plots (Friedman, 2001), the Local Interpretable Model-agnostic Explanations (LIME) (Ribeiro et al., 2016) and SHapley Additive exPlanations (SHAP) (Lundberg and Lee, 2017).

Gradient boosting is one particularly interesting Machine Learning algorithm. This model iteratively fits a new base model on the errors of the previous model. Then the results of the new base model and of the previous model are combined to create a new model. In this thesis, a gradient boosting algorithm called eXtreme Gradient Boosting (XGBoost) (Chen and Guestrin, 2016) is used. In a quick comparison with other gra-

dient boosting algorithms, XGBoost has a higher performance and is relatively easy to implement. As far as we know, Gradient Boosting has not yet been applied in a LGD setting.

Thus, firstly, this thesis explores whether the XGBoost model, considered as a black-box model, outperforms in terms of LGD prediction the natural interpretable models considered, namely the Fractional Response Model (Papke and Wooldridge, 1996) and the Regression Tree (Breiman et al., 1984).

Secondly, it aims to assess whether the use of interpretation techniques, namely Permutation Feature Importance, Partial Dependence Plots, and Shapley Additive Explanation, can help understand the inner workings of the XGBoost model. In the credit risk area, some empirical studies have already focused on the explainability of black-box models (Bracke et al., 2019). However, these studies focused on the probability of default and credit scoring. Applying the interpretation techniques presented above in the specific field of LGD study is expected to contribute to a better understanding of these techniques and how they can be applied in the case of LGD. Thus, if the above techniques are able to provide a better understanding of black-box models, Financial Institutions might be able to rely on black-box models and estimate losses more efficiently, without harming the interpretability of the applied model. Thus, they can gain a competitive advantage through the use of machine learning models to estimate LGD.

The growing interest in the use of black-box models initiated a new area of investigation of interpretability techniques. Therefore, it is expected that this study will trigger new empirical studies that involve applying these techniques to support the work already developed in the area and test its application in real data.

To address the objectives identified above, the remaining of the work is divided as follows: in the next section, a literature review is presented regarding the LGD models already developed, the main risk drivers considered to describe LGD, and the main existing interpretability approaches. In the third chapter, the defined methodology framework is presented, where the theoretical foundations of the considered models are presented. In chapter 4, a descriptive analysis of the data is made. In chapter 5, the results obtained are analyzed. Finally, the main conclusions to be drawn from this work are presented.

2. Literature review

2.1 Modeling of Loss Given Default

The first published studies related to LGD started in the '90s, focusing on the analysis of its distribution (Asarnow and Edwards, 1995). Later on, like other study areas, LGD began being analyzed by estimating a linear least squares regression (Acharya et al., 2007;

Caselli et al., 2008; Davydenko and Franks, 2008 and Grunert and Weber, 2009). However, these results can be questionable for two main reasons: (i) they do not guarantee that predictions fall within the unit interval as expected, taking into account the limited nature of the dependent variable; (ii) they ignore the non-constant partial effect of explanatory variables. Dermine and Neto de Carvalho (2006) overcame these drawbacks by using the fractional regression method (Papke and Wooldridge, 1996). Another way of overcoming these disadvantages is to properly transform the dependent variable before applying the linear model. An example of this is Gupton and Stein (2005), which, based on a sample of 3.026 defaulting instruments from 1981-2004, developed a calculation approach that consists of normalizing recovery rates via beta distribution and models using linear regression of variables (Moody's LossCalc v2).

Using Moody's Ultimate Recovery Database, Altman and Kalotay (2014) show that mixtures of Gaussian distributions, explicitly conditioned on borrowers characteristics, debt instrument characteristics and credit conditions at the time of default, yields more accurate forecasts of ultimate recoveries on portfolios of defaulted loans and bonds, on an out-of-sample basis, than popular regression-based estimates.

Recently, the first studies with the application of machine learning models to predict Loss Given Default began to appear. In most cases, it was concluded that the machine learning techniques outperform the usual parametric models (Bastos, 2010; Qi and Zhao, 2011; Loterman et al., 2012; Bastos, 2014; Yao et al., 2015).

Bastos (2010) starts from a database of 374 loans granted by a Portuguese Bank to small medium-sized enterprises (SMEs) that went into default between 1995-2000 to assess the capacity of a parametric fractional response regression and a nonparametric regression tree model to predict Loss Given Default. The results suggest that regression trees are an interesting alternative to parametric models in predicting and modeling this parameter.

Using the Moody's URD database, Qi and Zhao (2011) compare six modeling methods for Loss Given Default: four parametric methods (OLS regression, fractional response regression, inverse Gaussian regression, and inverse Gaussian regression with beta transformation) and two nonparametric methods (regression tree and neural network). They concluded that the nonparametric methods perform better than the parametric methods.

Loterman et al. (2012) present a large-scale benchmark study using 24 regression techniques that were evaluated on six real-life data sets obtained from major international banking institutions. The authors concluded that non-linear techniques, support vector machines, and artificial neural networks, in particular, produce significantly better model performances than more traditional linear techniques. These results suggest the presence of non-linear relationships, unlike previous benchmarking studies on the probability of

default (PD) modeling, where the differences between linear and non-linear techniques were not so explicit.

Using Moody's URD data, Bastos (2014) shows that the recovery rate predictions provided by a set of regression trees outperform the predictions provided by a single regression tree. The author warns of a reduction in this approach's explainability compared to traditional approaches, making it difficult to perceive which variables contribute to the improved forecasts.

Yao et al. (2015) use support vector regression (SVR) techniques to predict the Loss Given Default of corporate bonds and compare the results obtained with thirteen other algorithms. The authors conclude that the support vector regression techniques are a promising technique for banks to predict losses in the event of default.

2.2 Risk drivers of Loss Given Default

This section presents the supporting literature for identifying the most relevant variables related to the Loss Given Default. In the next subsections, the four main variables identified in relevant literature are explored: Industry, Collateral, Seniority and Priority in the liability structure.

2.2.1 Industry

The counterparty industry is one of the most frequent explanatory variables for the estimation of Loss Given Default. For corporate bonds, Altman and Kishore (1996) find evidence that a large number of sectors have similar recovery rates, but that there are considerable differences in some sectors, namely public services, which have considerably higher recovery rates than other sectors. Other studies such as Grossman et al. (2001), Dermine and Neto de Carvalho (2006) and Acharya et al. (2007) corroborate with the importance of the industry type as a main driver of LGD. On the other hand, Gupton et al. (2000), Franks et al. (2004) and Bastos (2014) detect no significant impact of industry type on recovery rate prediction.

2.2.2 Collateral

Another variable widely considered in the literature and which appears to be a determining factor of LGD is the existence and quality of collaterals associated with defaulted instruments (Asarnow and Edwards, 1995; Gupton et al., 2000, 2005 and Dermine and Neto de Carvalho, 2006). Not surprisingly, the results of these studies demonstrate that there is a positive effect between the existence of a collateral and the recovery rate.

2.2.3 Seniority and Priority in the liability structure

There is strong empirical evidence that seniority and debt cushion have a substantial impact on recovery rates. For example, for North American corporate issuers over 21 years (1983-2003), Varma and Cantor (2005) conclude that seniority and debt cushion are two of the three most important variables for determining the recovery rate. Regarding seniority, they concluded that the securities designated as "senior" tend to have higher recoveries than securities designated as "junior". Regarding debt cushion, they concluded that the greater the value of the debtor's junior debt compared to the instrument's debt (in terms of total debt), the greater the amount that is expected to be recovered, leading to higher expected recovery rates. These results have economic plausibility because the larger the debt cushion, the greater the amount that is likely to be available for distribution to more senior applicants. Thus, it is not surprising that there is a positive and statistically significant coefficient between the debt cushion variable and the recovery rate prediction.

3. METHODOLOGY

3.1 Modeling techniques

3.1.1 Natural Interpretable Models

According to Miller (2019), interpretability is defined as the degree to which a human can understand the cause of a decision. In this way, it is considered as Natural Interpretable Models the models in which the decision-making processes based on the estimation of these models are easier for a human being, requiring no effort in the implementation of additional methods to support decision making. Using Natural Interpretable Models is the easiest way to obtain interpretability. Some examples of this Natural Interpretable Models are linear regression, logistic regression, and the decision tree. Taking into account the limited nature between 0 and 1 of the dependent variable (see Figure 4.1), it is not recommended to implement a regression of ordinary least squares (OLS), since it does not guarantee that the predicted values always fall between the desired range. Thus, the following models were selected as Natural Interpretable Models: (1) the Fractional Response Model (FRM) and (2) the Regression Tree (RT). In this section, the theoretical background of each of these models is introduced.

Fractional Response Model

The FRM is a glass-box parametric model. Because recovery rates are bounded to [0, 1] we want to estimate a parametric model suited for modeling fractional response

variables. The model is

$$E(r|X) = G(\beta_0 + \beta_1 x_1 + ... + \beta_k x_k) = G(X\beta), \tag{3.1}$$

where $G(.)$ satisfies $0 < G(z) < 1$ for all $z \in \mathbb{R}$. This condition ensures that the predicted values fall within the unit interval. There are several functional forms for $G(.)$, the most common being the cumulative normal distribution, the logistic function, and the log-log function. The logistic function was selected as the functional form of $G(.)$:

$$G(X\beta) = \frac{1}{1 + \exp(-X\beta)}, \tag{3.2}$$

The non-linear estimation procedure consists of the maximization of the Bernoulli log-likelihood function (Papke and Wooldridge, 1996):

$$l_i(\beta) \equiv r_i \log[G(x_i\beta)] + (1 - r_i) \log[1 - G(x_i\beta)], \tag{3.3}$$

The quasi-maximum likelihood estimator (QMLE) is consistent and asymptotically normal, regardless of the distribution of the recovery rate r_i conditional on x_i.

Since the function $G(x)$ is non-linear, the partial effects of the explanatory variables on the recovery rates are not constant. They can be calculated for specific values of the explanatory variables. Thus, the partial effect of variable x_j on the recovery rate is given by:

$$\frac{\partial E(r|X)}{\partial x_j} = \frac{dG(X\beta)}{d(X\beta)}\beta_j, \tag{3.4}$$

Because $G(X\beta)$ has a strictly monotonic behavior, the coefficient's sign provides the direction of the partial effects. For discrete values, the loan recovery logit function, $G(X\beta)$, is calculated with and without activating the flag. The partial effect is then calculated as the relative increase in the rate of recovery of defaulted instruments with and without the flag's activation.

Regression Tree

Regression tree is a non-parametric forecasting model (unlike the fractional response model presented in the previous section) and non-linear where the original data set is recursively partitioned into smaller data sets using a greedy search algorithm.

Starting with a root node containing all available observations, the algorithm searches all possible binary splits, using all regressors, which minimizes the intra-subset variation of the target variable in the newly created child nodes, making sure that in each child node, the target variable is more homogeneous than in the parent node. This procedure

is repeated recursively for the new child nodes until there is no further reduction in the variation of the target variable. Unsplit terminal nodes are denoted by 'leaves'. Figure A.1 in the Appendix, illustrates the typical structure of a regression tree. The decrease in the variance of the target variable is measured by the 'Sum of squares reduction',

$$SSRe = SST - (SSL + SSR), \tag{3.5}$$

where $SST = \sum_{i=1}^{n}(y_i - \bar{y})^2$ is the sum of squares for the node and SSR and SSL are sums of squares for the right and left child, respectively. All observations start at the root node, follow a path and end on a leaf. This model's structure ensures that the prediction is limited between 0 and 1 since the predicted values are equal to the average of the target variable (i.e. recovery rate).

As its name implies, this algorithm is greedy, very often resulting in extremely large and complex trees that over-adjust the data, resulting in excellent predictions when applied to the training set but in wrong predictions when applied to the test set (this is known as overfitting). Thus, the application of mechanisms such as pruning is necessary to avoid this problem. After the tree grows freely, the pruning procedure reduces the forecast tree produced to a smaller tree using a cost complexity parameter[1].

In addition to adjusting to the nature of the target variable, the regression trees are not affected by outliers' presence, since they are isolated in a node and have no more effect on the division. In this study's scope, the main advantage that we can derive from regression trees is that they create good explanations since the tree structure automatically invites us to think of the values provided for individual observations as counterfactual. One does not need to be an expert in modeling techniques to, when looking at a regression tree, arrive at the logical understanding that "If a regressor had been greater/less than the dividing point, the prediction would have been y_1 instead of y_2". The explanations of the tree are contrasting since it is possible to compare the predictions of an instance with the relevant "and if" scenarios (as defined by the tree), which are simply the other nodes of the leaves of the tree.

3.1.2 Black-Box Model - eXtreme Gradient Boosting (XGBoost)

This section presents the theoretical foundations to support the selected black-box model. Extreme Gradient Boosting, known as XGBoost, has reinvented the previously existing tree boosting algorithms. Since 2015, it has been widely recognized in several machine learning and data mining challenges. In the winning solutions posted on Kaggle during 2015, about 60% used XGBoost. The success of XGBoost also went through

[1]For more details on the pruning process used, see: `https://cran.r-project.org/web/packages/rpart/rpart.pdf`

the KDD Cup, where, also in 2015, all the top 10 solutions used XGBoost (Chen and Guestrin, 2016).

Gradient descent

A gradient descent algorithm is an algorithm that minimizes a loss function. Suppose that we have observation pair $z = (x, y)$ and a function $F_w(x)$, with parameters w, mapping explanatory variables x to observation y and that a differentiable loss function $L(z, w)$ is defined to model the performance of $F_w(x)$. The main idea of the gradient descent is to use the gradient of this loss function with respect to the different function parameters or weights w to find the values of w that minimize the selected loss function. For initialization, the starting values of w (w_0) need to be defined. In the base case, called Batch Gradient Descent, the gradient's value is calculated at each available observation pair z_i. After this, all derivatives are averaged, and consequently, the new weight is calculated. This process is performed iteratively as follows,

$$w_{t+1} = w_t - \gamma \frac{1}{n} \sum_{i=1}^{n} (\nabla_w L(z_i, w_t)), \tag{3.6}$$

where w_{t+1} are the optimal parameters found using the derivative of the loss function $\nabla_w L(z_i, w_t)$, γ is the learning rate and is always positive and z_i is the observation pair (x_i, y_i) for observation i. Dennis and Schnabel (1996) show that when the learning rate is small enough, this iterative process converges to a local minimum.

Boosting

Boosting is an ensemble method. First introduced by Kearns (1988, 1989), and with a strong contribution from Schapire (1990), ensemble methods are based on the idea that a set of weak individual learners can be combined to result in a better output that reduces the generalization error of the prevision. Unlike bagging, where additive models are formed simultaneously and every model gets an equal vote, boosting is processed sequentially. The subset of training data used in each member of the ensemble is selected based on the performance of the previous model. Observations that were predicted incorrectly by previous models are chosen more frequently to enter the estimation dataset of the next member than observations that have already been correctly predicted.

For a given data set with n observations and m regressors $D = (x_i, y_i)(|D| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R})$, a tree ensemble model uses K additive functions to predict the output.

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F, \tag{3.7}$$

here, where $F = f(x) = w_{q(x)}(q : \mathbb{R}^m \rightarrow T, w \in \mathbb{R}^T)$ is the space of regression trees, q represents the structure of each tree that maps an observation to the corresponding leaf index and T is the number of leaves in the tree. Each f_k corresponds to an independent tree structure q and leaf weights w.

XGBoost algorithm

In this section, the details of the eXtreme Gradient Boosting (XGBoost) algorithm introduced by Chen and Guestrin (2016) for the regression problem are explained. The loss function used in XGBoost is a combination of the mean squared error loss function and a penalty term Ω. The mean squared error calculates the predicted recovery rate error, $\hat{y}_1$, using the actual recovery rate. The penalty term Ω protects regression trees (weak learners) against over-fitting on the data by penalizing for complexity. Because of this penalty term for complexity, this loss function is also called the *regularized loss function*,

$$L(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k), \tag{3.8}$$

where $\Omega(f) = \gamma T + \frac{1}{2}\lambda||w||^2$ and $l(\hat{y}_i, y_i) = (y_i - \hat{y}_i)^2$

Here, T is the number of leaves in the tree, while each $f(k)$ corresponds to an independent tree structure q and leaf weights w. l is a differentiable convex loss function (mean squared error). Additionally, γ and λ are the regularization parameters. Note that the regularization loss function ($L(\phi)$) only reduces when a new tree is added to the model if the added model complexity of adding another tree does not exceed the added value of the new tree model and hence, recovery rates are modeled better. Let $\hat{y}_i^t$ be the predicted recovery rate of the i-th instance at the t-th iteration. The objective is to add the tree f_t that minimizes the loss function, formally

$$L^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t), \tag{3.9}$$

where f_t that most improves the model according to 3.8 is greedily added. XGBoost uses the second order approximation to optimize Equation 3.9,

$$L^{(t)} \simeq \sum_{i=1}^{n} [l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t), \tag{3.10}$$

where $g_i = \frac{\delta}{\delta \hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$ is the first order derivative of loss function l and $h_i = \frac{\delta^2}{\delta^2 \hat{y}_i^{(t-1)}}{}^2 l(y_i, \hat{y}_i^{(t-1)})$ is the second order derivative of the loss function l at every event of default i, both with respect to the predicted recovery rate of the previous iteration $\hat{y}_i^{(t-1)}$.

The objective of each iteration t is to find the regression tree f_t that minimizes this loss function. Hence, the terms that do not depend on this new tree, f_t, can be removed from the objective function.

$$\tilde{L}^{(t)} = \sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \Omega(f_t), \tag{3.11}$$

The main difference between the boosting process of XGBoost and the Gradient boosting method, introduced by Friedman (2001), is that while the Gradient boosting is based on first-order gradient descent updates, the XGBoost algorithm is based on second-order gradient descent updates. Sigrist (2018) shows that Newton boosting (used in the XGBoost algorithm) performs significantly better than the other boosting variants for regression problems.

Taking into account the objective function defined previously, the second procedure of interest is to find the split points that split the tree into leaves and branches. Define $I_j = i|f_t(x_i) = j$ as the set of observations i in leaf or terminal node j of the tree t. Equation 3.11 can be rewritten by expanding Ω and summing over the instances of each leaf $j, i \in I_j$, where j are the possible leaves, as

$$\begin{aligned} \tilde{L}^{(t)} &= \sum_{i=1}^{n} [g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i)] + \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \\ &= \sum_{j=1}^{T} [(\sum_{i \in I_j} g_i) w_j + \frac{1}{2} (\sum_{i \in I_j} h_i + \lambda) w_j^2] + \gamma T, \end{aligned} \tag{3.12}$$

where the second part follows from separating all observations i in observation sets I_j and the value for $f_t(x_i)$ of observation i in observation set I_j is w_j. The derivatives g_i and h_i are constant, since they depend on the loss function of the previous prediction and hence do not depend on the proposed tree f_t. For a fixed tree structure $q(x)$ with known observation sets I_j in leaf j, the optimal value for weight w_j^* in leaf j which is the optimal recovery rate, can be found by solving the quadratic equation in 3.12 for w_j.

$$w_j^* = -\frac{\sum_{i \in I_j} g_i)}{(\sum_{i \in I_j} h_i + \lambda)}, \tag{3.13}$$

By substitution in 3.13, the minimum value for the loss function can be given as

$$\tilde{L}^{(t)}(f_t) = -\frac{1}{2} \sum_{j=1}^{T} \frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T. \tag{3.14}$$

The previous equation can be used to assess the loss reduction of a particular split. Assume that L_L and L_R are the instance sets of left and right nodes after a split where $I = I_L \cup I_R$. Then, the loss reduction after the split is given by

$$L_{split} = \frac{1}{2}[\frac{(\sum_{i \in I_L} g_i)^2}{\sum_{i \in I_L} h_i + \lambda} + \frac{(\sum_{i \in I_R} g_i)^2}{\sum_{i \in I_R} h_i + \lambda} - \frac{(\sum_{i \in I} g_i)^2}{\sum_{i \in I} h_i + \lambda}] - \gamma. \tag{3.15}$$

Therefore, it is only necessary to add the gradient and the statistics of the second-order gradient on each leaf and then to apply the scoring formula (3.14) to obtain the quality score (Chen and Guestrin, 2016).

Hyperparameter Optimization

Hyperparameters refer to properties of the model that cannot be directly learned in the regular training process. Thus, they must be defined before starting the learning process. The performance of a model can increase significantly with the right set of parameters. Hence an important aspect in modeling with XGBoost is tuning the hyperparameters.

For the optimization of the parameters, the Grid Search methodology was used. This entails setting up a grid of many possible hyperparameters and creating a model with all pre-specified hyperparameters using cross-validation. Table A.1 of the Appendix presents a table with the description and the proposed settings for the hyperparameters targeted for tuning for this study, namely: Learning rate, Minimum child weight, Maximum tree depth, Column sample percentage, Subsample percentage, and Gamma.

3.2 Selected approaches to interpretability

It is possible to divide the interpretability approaches into two types: model specific and model agnostic. A model specific approach applies to a restricted set of models, such as a linear model's regression weights. On the other hand, model agnostic methods can be applied to any model, usually involving the parallel analysis of input regressors and output values. Additionally, an interpretation method can be applied at the global or local level (Guidotti et al., 2018). Interpreting a model globally means that interpretation is performed so that the functional form of the trained model is perceptible to a human (Yang et al., 2018). Interpreting a model locally is related to focusing on a single observation and examining what the model predicts for a specific input (Molnar, 2019). Within academic literature, the following methods are commonly used.

The Global surrogate method (Craven and Shavlik, 1996) is based on the construction of simple models (such as linear regression or simple decision trees) that approximate the functioning of a complex model. One way of measuring how well the surrogate model

replicates the complex model is the R-squared measure, which measures the percentage of variance captured by the surrogate method. After selecting the surrogate model by obtaining a satisfactory R-square measure, the simple model is interpreted. The conclusions about the complex model are made based on the interpretation of the simple model. This approach is easy to implement and explain to people unfamiliar with Machine Learning. However, it only allows conclusions to be drawn about the black-box model and not about the data, since the surrogate models are trained only using the predictions of the black-box model instead of being trained with the observed values.

For models based on decision trees, there is an interpretability technique that includes feature importance measures, namely considering the frequency of selection of variables in the models and the accuracy gained by the model due to the use of a given variable.

Another option to measure the global importance of variables that is not only applicable to models based on decision trees is to use the Permutation Feature Importance method (Fisher et al., 2018), which measures the change in the forecast error after the permutation of the values of the regressors breaking the relationship between the regressor and the output. This method offers a global insight into how the model works. However, since it is linked to the model error, it does not allow us to identify the expected variation in the output due to a certain feature's permutation.

The Partial Dependence Plot (PDP) (Friedman, 2001) shows the impact that one or two variables have on the predictive outcome. This tool allows observing non-linear relationships between the target variable and a regressor. The main disadvantage of this method is that it assumes independence between the variables considered in the construction of the PDP and the other regressors.

Local Surrogate Models approximate the complex model's predictions on selected sub-sections of the data. The most used model of this type is the Local Interpretable Model-agnostic Explanations (LIME) introduced by Ribeiro et al. (2016). Instead of training a global surrogate model, LIME focuses on training local surrogate models to explain individual predictions.

Finally, the SHapley Additive exPlanations (SHAP) introduced by Lundberg and Lee (2017) is a method based on cooperative game theory and Shapley values. While LIME only gives local approximations, the SHAP method provides globally consistent explanations. The final prediction is broken down into contributions from each regressor, making the sum of all contributions equal to the final prediction.b

3.2.1 Classical Feature Importance

The three measures that are possible to find in any tree-based modeling package are considered as classical feature importance metrics: (1) Coverage; (2) Frequency; and (3)

Gain.

Coverage considers the number of times a regressor is used to split data across all trees weighted by the number of training data points that pass through these divisions. For example, having 200 observations, six regressors, four trees, and assuming that $regressor\_A$ is used to decide the leaf node for $10, 5, 6$, and 2 observations on $tree1$, $tree2$, $tree3$, and $tree4$ respectively; then, the metric will count the coverage for this feature as $10 + 5 + 6 + 2 = 23$ observations. This will be calculated for all six regressors, and coverage will be expressed as a percentage of all regressors' coverage metrics.

The Frequency considers the number of times a regressor is used to split data across all trees. In the above example, if $regressor\_A$ occurred in $2splits, 1split, 4splits$, and $3splits$ in $tree1$, $tree2$, $tree3$, and $tree4$, then the Frequency for $regressor\_A$ will be $2 + 1 + 4 + 3 = 10$. The Frequency for $regressor\_A$ is calculated as its percentage frequency over frequencies of all features.

Finally, Gain takes into account the average reduction in training loss achieved when using a split feature. When compared to another regressor, a higher value for this metric implies that the regressor is more important for generating a prediction.

3.2.2 Permutation Feature Importance

The Permutation Feature Importance approach measures the increase in the model's prediction error after permutation of the regressors values, which breaks the relationship between the regressors and the true result. This method considers a regressor "important" if, after shuffling its values, the model error increases and "unimportant" if the model error does not vary. If the model error increases, after shuffling, it means that the model relied on that specific regressor for the prediction. On the other hand, if the error does not change, it means that the model ignored the regressor for the prediction. Breiman (2001) introduced the permutation feature importance measurement for random forests. Based on this idea, Fisher et al. (2018) proposed "model reliance", a model-agnostic version of the feature importance. In Table A.2, the pseudocode for the importance algorithm of the permutation regressor based on Fisher et al. (2018) is presented. The loss function considered was the Mean Absolute Error (MAE).

3.2.3 Partial Dependence Plots (PDP)

The metrics used to quantify the importance of the regressors presented above do not allow for the identification of the type of relationship between the regressors and the output. An effective approach for explainning the output that results from black-box models is to use partial dependency plots (PDP) introduced by Friedman (2001). The PDPs show the marginal effects of a reduced number of regressors (usually 1 or 2) in

predicting the model's output, enable to identify whether the relationship between the regressor and the output is linear, monotonic, or more complex. When applied to the linear regression model, the resulting plots would be a simple straight line whose slopes are equal to the model parameters for each feature.

Following mostly Friedman (2001) and Greenwell et al. (2001), Partial Dependence Plots are introduced below. Let $x = x_1, x_2, ..., x_p$ represent the features in a model whose prediction function is $\hat{f}(x)$. Partitioning x into an interest set, z_s, and its complement, $z_c = x \setminus z_s$, then the "partial dependence" of the response on z_s is defined as

$$f_s(z_s) = E_{z_c}[\hat{f}(z_s, z_c)] = \int \hat{f}(z_s, z_c) p_c(z_c) dz_c, \tag{3.16}$$

where $p_c(z_c)$ is the marginal probability density of $z_c : p_c(z_s) = \int p(x) dz_s$. Equation 3.16 can be estimated from a set of training data by

$$\bar{f}_s(z_s) = \frac{1}{n} \sum_{i=1}^{n} \hat{f}(z_s, z_{i,c}), \tag{3.17}$$

where $z_{i,c}(i = 1, 2, 3, ..., n)$ are the values of z_c that occur in the training sample. That is, the average of the effects of all other features of the model.

In this way, the construction of the PDPs is quite straightforward. Let $z_s = x_1$ be the predictor variable of interest with the value vector $x_{11}, x_{12}, ..., x_{1k}$. The partial dependence plot of the response on x_1 can be constructed according to the following pseudocode:

Input:
the unique predictor values $x_{11}, x_{12}, ..., x_{1k}$
Process:
for $i \in 1, 2, ..., k$
copy the training data and replace the original values of x_1 with the constant x_{1i}
compute the vector of predicted values from the modified copy of the training data
compute the average prediction to obtain $\bar{f}_1(x_{1i})$
Output:
the estimated partial dependence values $f_1(x_{11}), f_1(x_{12}), ..., f_1(x_{1k})$

TABLE 3.1: Pseudocode for constructing the partial dependence of the response on a single feature x_1. Based on Greenwell et al. (2001)

The PDP for x_1 is obtained by plotting the pairs $x_{1i}, \bar{f}_1(x_{1i})$ for $i = 1, 2, ...k$. For a better understanding, the attached Figure A.2 presents a simplified representation of the process described above.

For categorical variables, partial dependence plots are calculated using the estimates by forcing all data instances to have the same category. For example, for the partial dependence plot applicable to the variable $Never\_defaulted$, two numbers would be calculated: having previously been defaulted ($Never\_defaulted = 0$) or never having been defaulted previously ($Never\_defaulted = 1$). To calculate the value for the first

case, one needs to set the defaulted flag of all data instances with "0" and average the predictions.

Partial dependence plots are easy to implement and interpret: the partial dependency function on a specific regressor value represents the average forecast if all data points are forced to assume that regressor value. However, they are not perfect in specific circumstances. Assumption of independence is a major problem for PDPs since it is assumed that the feature for which the partial dependence is computed is not correlated with other features. In the case of strong interactions or a high correlation between regressors, the PDP's output value might be biased and lead to wrong causal interpretations. Another disadvantage of the PDP is related to the heterogeneous effects that can be hidden once the PDP shows only the average marginal effects. The positive and negative effects can be canceled, and the PDP can pass the wrong conclusion that a feature has no effect on the prediction.

3.2.4 SHAP (SHapley Additive exPlanations) Values

Shapley Additive exPlanations also called SHAP Values (Lundberg and Lee, 2017), is a model agnostic method of explaining a model.

Additive feature attribution methods

SHAP belongs to the class of models called "additive feature attribution methods", where the explanation is expressed as a linear function of features. The explanation model f can approximate output z' with the attribution value of each feature ϕ_t.

$$f(z^{'}) = \phi_0 + \sum_{t=1}^{M} \phi_t z_t^{'}, \tag{3.18}$$

where $\phi_t \in \mathbb{R}, z^{'} \in \{0,1\}$ and M is the number of interpretable input features. $z_t^{'}$ represents the appearance of the features. If the feature is observed then $z_t^{'} = 1$. Otherwise, $z_t^{'} = 0$. Instead of the original feature, SHAP replaces each feature (x_i) with a binary variable $(z_t^{'})$ that represents whether (x_i) is present or not.

Shapley Values

The basis of the SHAP method are the Shapley values (Shapley, 1953) of game theory. Applying to machine learning predictions and interpretability, the "game" is the prediction task for a single dataset observation. The "gain" is the current prediction for this instance minus the average prediction for all observations. The "players" are the regressors' values of the observations that collaborate to receive the gain. Thus, the added value of a feature

as the weighted increase or decrease in the value of a model outcome can be calculated when adding a feature i over all subsets of features that exclude that feature ($S \subseteq F \setminus i$). The SHAP values for features in a model indicate the attribution of those features to the prediction outcome. The SHAP value for a feature i and model f is calculated by

$$\phi_i(f) = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F| - |S| - 1)!}{|F|!} (f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S)), \tag{3.19}$$

where F is the total set of features for model f, $S \subseteq F \setminus \{i\}$ are all possible subsets of F excluding feature i, $f_S(x_S)$ is the function trained on features of subset S and the same holds for function $f_{S \cup \{i\}}(x_{S \cup \{i\}})$ on set $S \cup \{i\}$.

SHAP Feature Importance

To calculate a measure of feature importance, SHAP considers as most important the features with the highest absolute Shapley value. Since the objective is to measure the global importance of the features, it considers the average of the absolute Shapley values per feature over the entire dataset.

$$I_i = \sum_{j=1}^{n} |\phi_i^{(j)}|, \tag{3.20}$$

where i represents each feature and j each observation of the dataset. The feature importance graph is given by the representation of decreasing order of the variables importance.

SHAP feature importance is an alternative to permutation feature importance (see section 3.2.2). However, there is a big difference between the two measures of importance. While the importance of the permutation feature importance approach is based on the decrease in the model's performance, the SHAP's importance is based on the magnitude of the contributions of its regressors.

3.3 Sampling: training and test data

This section presents the selected validation model for measuring the predictive accuracy of the models. To assess the behavior of the model in the presence of new observations, most of the existing methods are based on the use of two types of distinct data sets: a dataset used to train the model (Training Set) and a test set, used to test the model (Test Set).

A very straightforward and widely used method is the holdout method. This method only involves splitting the dataset into two sets: 'train' and 'test' data. A common split used in this method is to randomly separate 80% of the dataset as a training sample and

20% as a test sample (see Figure A.3). The model will be executed looking only at the training set and, after that, the test set is used to evaluate its performance. This method depends only on a split between test and training set, making it highly dependent on the cutting point. In addition to wasting a large part of the observations in the model's training process (i.e. 20%), this method needs a reasonably large test set to accurately assess the prediction error.

The traditional approach to address this problem is to resort to k-fold cross-validation (Kim, 2009), where observations are broken into k sets of equal size. In the first call, the model is estimated using all subsets except the first (called the first fold). The held-out subset is predicted by this model and used to estimate performance measures. After this, the first subset is included in the training set and the procedure repeats with the second subset held out, and so on. In this way, all observations are used both in the training and test sets. The error estimate is averaged across all k tests to obtain the total accuracy of the model. The cross-validation process with $k = 5$ is depicted in Figure A.4.

As k gets larger, the difference between the estimated and true values of performance (bias) becomes smaller (i.e., the bias is smaller for $k = 10$ than $k = 5$). An unbiased method may be estimating the correct value but may pay a high price in uncertainty (i.e., variance). This means that repeating the resampling procedure may produce a very different value. Thus, the objective is to maximize the model's precision (bias) and minimize its complexity (variation). Molinaro (2005) found that the leave-one-out [2] and k-fold cross-validation with $k = 10$ yielded similar results, indicating that $k = 10$ is more attractive from the perspective of computational efficiency. To develop models with a large fraction of the available data and evaluate the predictive accuracy with the complete dataset, a 10-fold cross-validation was implemented. For hyperparameters optimization, a $k = 5$ was considered.

3.4 Evaluating predictive accuracy

The expected predictive accuracy of the developed models on new data (Test Set) is assessed using 5 performance measures widely used in the literature of recovery rate estimation (Bastos, 2014 and Dermine and Neto de Carvalho, 2006).

Let y and $\hat{y}$ denote the actual and predicted recovery rates, respectively, and n denote the number of defaulted events in the sample. The mean squared error (MSE) is defined as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{3.21}$$

Models with a high MSE tend to show greater differences between actual and predicted

[2] K-fold cross-validation process where k is equal to the number of observations.

recoveries, meaning that, on average, they predict real recoveries with less accuracy. On the other hand, models with low MSE have lower differences, thus predicting more accurately.

The mean absolute error (MAE) is defined as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|, \tag{3.22}$$

Since the recovery rate is a ratio (between 0 and 1), the MAE can reflect the size of the error in a more intuitive and direct way. Models with lower MAE also predict actual values more accurately, on average. By using squares, unlike MAE, the MSE places more weight on large errors than on small ones. Both the MSE and the MAE simply look at the average difference between actual and predicted recovery rates. In this way, the interpretation of these error measures is done on the scale of the variable of interest. The relative absolute error (RAE), expressed as a percentage, is defined as:

$$RAE = 100\frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{\sum_{i=1}^{n}|y_i - \bar{y}|}, \tag{3.23}$$

The relative squared error (RSE), expressed as a percentage, is defined as:

$$RSE = 100\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}, \tag{3.24}$$

Relative errors measure the predictive accuracy with respect to the average outcome of the response variable, thereby neglecting the information provided by the explanatory variables. Models with RSE and RAE smaller than 100% provide, on average, better predictions than the simple predictor ($\bar{y}$) in terms of squared and absolute error, respectively. The statistical correlation between y and $\hat{y}$ is defined as:

$$\rho_{y,\hat{y}} = \frac{Cov(y,\hat{y})}{\sqrt{Var(y)Var(\hat{y})}}, \tag{3.25}$$

Naturally, models with a higher correlation have a better performance when compared with models that display a lower correlation.

4. DATA

4.1 Moody's Ultimate Recovery Database

For this research, we used Moody's Ultimate Recovery Database (Moody's URD), which covers US non-financial corporations holding over $50 million in debt at the time

of default. The data set contains 4630 defaulted instruments (bonds and loans) from 957 different obligors covering default events between 1987 and 2010. Moody's URD includes three different valuation methods for nominal recoveries (settlement method, trading price method, and liquidity event method) and indicates for each defaulted instrument which of the methods is more representative. This study uses the discounted recovery rate associated with the valuation method recommended by Moody's for each defaulted instrument.

Figure 4.1 shows the distribution of discounted recovery rates. It can be observed that the distribution is bimodal with a greater incidence on the right. Approximately 20% and 40% of the instruments have complete or almost complete loss (recovery rate less than or equal to 10%) or complete or almost complete recovery (recovery rate greater than 90%), respectively. The average RR on instruments included in the dataset is 59%. Through an analysis per default year (Figure A.5) it is clear that there is a cyclical effect

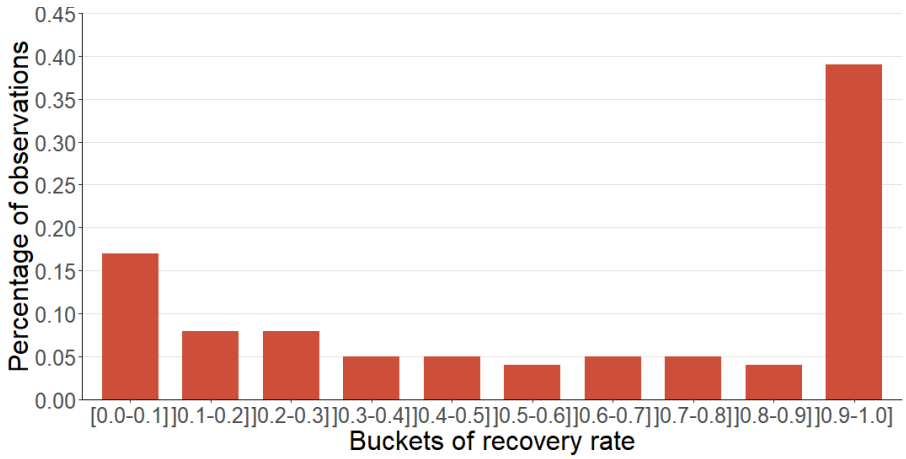

FIGURE 4.1: Distribution of discounted recovery rates for defaulted instruments in Moody's Ultimate Recovery Database (1987-2010)

of the number of defaulted instruments over time. There is an increase in the number of defaulted instruments in the recessions of the early 1990s, early 2000s, and late 2000s. The defaulted instruments that went into default in 1989 have an average recovery rate lower than the other years (46%), while defaulted instruments that went into default in 2005 have an average recovery rate higher than the remaining years (76%). There is a substantial variation in the average recovery rates over the entire observation period.

Moody's URD has been used in several academic studies (among which Qi and Zhao (2011) and Bastos (2014)), making it a reference dataset for the study of recovery rates.

4.2 Variables

This subsection presents the main characteristics of all the variables considered in this study. Also, during each variable's presentation, the necessary treatments that have been carried out are mentioned, namely the transformation into dummies of the categorical variables.

Industry

Table 4.1 shows the sample distribution and the mean recovery rate by Moody's industry classification. The lowest historical recovery rates occurred in the Environment industry (29%), Telecommunication industry (42%), and Construction industry (48%) while the highest occurred in the Natural products industry (82%) and the Energy industry (74%). Jointly Energy, Distribution, Telecommunications, Manufacturing and Consumer products industries represent close to 50% of total defaulted instruments. This shows great differences in the recovery rate across industries. For the inclusion of the industry in the models, dummy variables were generated, having the Telecommunications sector as a reference group.

Industry	Instruments	Average RR%	Industry	Instruments	Average RR%
Automotive	204	62%	Manufacturing	427	64%
Chemicals	74	64%	Media	358	64%
Construction	68	48%	Metals & mining	141	57%
Consumer products	385	65%	Natural products	93	82%
Distribution	519	52%	Other	67	57%
Energy	493	74%	Services	337	58%
Enviroment	51	29%	Technology	146	61%
Healthcare	157	55%	Telecommunications	469	42%
Industrials	69	67%	Transportation	314	50%
Leisure & entertainment	258	62%			

TABLE 4.1: Number of instruments and average recovery rate by industry in Moody's Ultimate Recovery Database (1987-2010)

Instrument type

Table 4.2 shows the sample distribution and the average recovery rate by type of instrument. Defaulted instruments can be separated into two large groups: bonds (about 60% of the dataset) and loans (about 40% of the dataset). Bonds have an average recovery rate of 45%, while loans have an average recovery rate of 80%, displaying much better recovery than bonds. This behavior reflects the typically higher credit position in terms of claims priority. The average recovery rates by type of instrument vary between 18% (Junior subordinated bonds) and 85% (Revolver loans). For the inclusion of the instrument

type variable in the models, dummy variables were generated, the reference group being the Junior subordinated bonds.

Instrument Type	Instruments	Average RR%
Junior Subordinated Bonds	69	18%
Revolver Loans	963	85%
Senior Secured Bonds	587	64%
Senior Subordinated Bonds	493	29%
Senior Unsecured Bonds	1263	49%
Subordinated Bonds	372	29%
Term Loans	883	76%

TABLE 4.2: Number of instruments and average RR by instrument type in Moody's Ultimate Recovery Database (1987-2010)

Collateral type

The sample breakdown by collateral type is shown in Table 4.3. As expected, unsecured instruments present the lowest mean recovery rate. Instruments secured by inventory accounts receivable and cash present higher recovery, which is also expected since these assets are easier to liquidate than other collaterals. For the inclusion of the collateral type variable in the models, dummy variables were generated, the reference group being the unsecured instruments.

Collateral Type	Instruments	Average RR%
All or most assets	1348	82%
Capital Stock	199	71%
Inventory, accounts receivable & cash	202	96%
Other	62	83%
Property, Plant & Equipment	342	59%
Second and third lien	204	55%
Unsecured	2273	41%

TABLE 4.3: Number of instruments and average recovery rate by collateral type in Moody's Ultimate Recovery Database (1987-2010)

Priority in the liability structure

Figure 4.2 shows the distribution of the average recovery rate by buckets of percentage above (the percentage of senior debt compared to defaulting instrument debt in the obligor's total debt) and percentage below (or "debt cushion", the percentage of junior debt compared to defaulting instrument debt in the obligor's total debt). This figure shows that debt position in the liability structure matters. The chart on the left shows that the

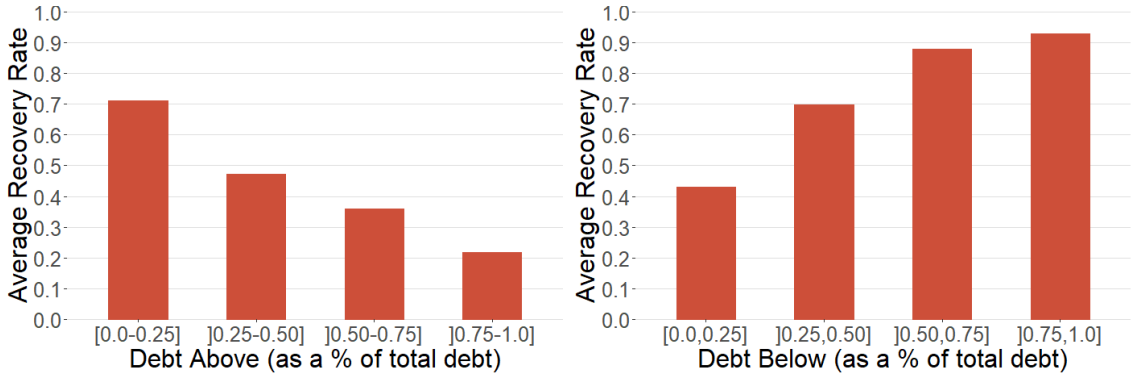

FIGURE 4.2: Average recovery rate by Debt above percentage and Debt below percentage in Moody's Ultimate Recovery Database (1987-2010)

average recovery rate declines as the percentage of total senior liabilities increases. Recovery rates average only 22% for those defaulted instruments with percentage of $above$ equal to at least 75% of total liabilities, compared to 71% for defaulted instruments with senior debt equal or less than 25% of total liabilities. On the other hand, as expected, the graph on the right shows that when the percentage of total junior debt increases the average recovery rate increases. Recovery rates average 93% when $cushion$ is greater than at least 75% of total liabilities, compared to an average of 43% for defaulted instruments with $cushion$ less than or equal to 25% of total liabilities. The variables $above$ and $cushion$ are continuous variables between zero and one. Additionally, it was also considered the ranking variable, which represents the debt's seniority in obligor liability structure where 1 corresponds to most senior and 7 to most junior. It is observed that the closer to 7 the ranking is, the lower the recovery rate. Debt rank is the only variable with a different scale, since it is measured in an integer scale ranging from 1 to 7. Because of this, the ranking variable was divided by 7.

Debt and previous default

Figure 4.3 shows the distribution of average recovery rate by buckets of instrument outstanding amount at default relative weight in obligor's total debt ($instdebt$) and never defaulted variable. The variable $never_defaulted$ is equal to "1" when the obligor's has never defaulted before and "0" otherwise. On the left side of Figure 4.3, it can be seen that, as the percentage of the outstanding amount as a percentage of the total debt increases, the average recovery rate decreases. On the right side, it is observed that for defaulted instruments belonging to obligors that had previously defaulted, the recovery rate is lower (close to 55%) than that of defaulting instruments belonging to obligors that have never defaulted (close to 100%).

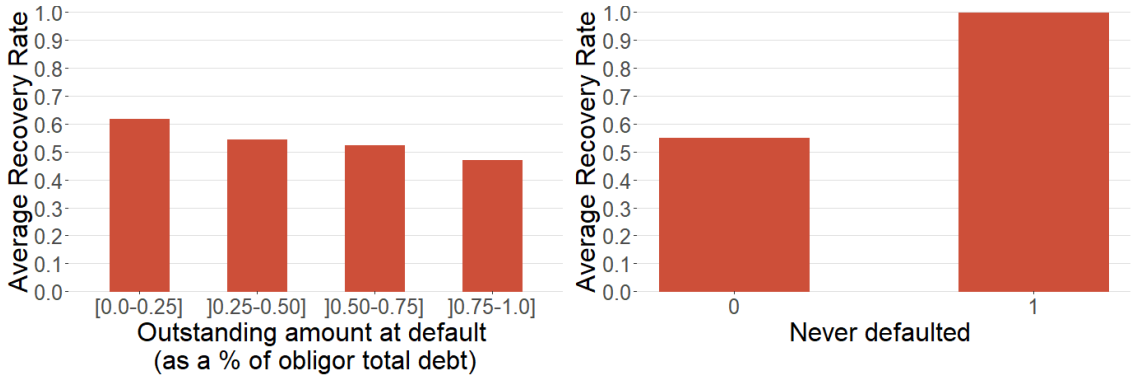

FIGURE 4.3: Average recovery rate by instrument outstanding amount (as a percentage of obligor total debt) and Never defaulted variable in Moody's Ultimate Recovery Database (1987-2010)

5. RESULTS

5.1 XGBoost - Hyperparameter tuning

Table 5.1 shows the hyperparameters that contribute to the best model according to the optimization technique considered (grid search). The performance gained in using these parameters is notorious, with a MAE 50% lower than the worst-performing parameter set.

Hyperparameter	Value
Learning rate	0.05
Max tree depth	14
Gamma	0
Sub sample	0.9
Column sample	0.7
Minimum child weight	1

TABLE 5.1: Hyperparameters used to fit final XGBoost model

5.2 Models performance comparison

Table 5.2 presents the out-of-sample accuracy measures for recoveries predicted by selected natural interpretable models (i.e. fractional response model and regression tree) and the black-box model (i.e. XGBoost). The last two columns show the percent variation of the accuracy measures between the fractional response model and a regression tree (FRM-RT), and between the regression tree and the XGBoost (RT-XGB). The same training and test samples are used to adjust and evaluate the accuracy of all models.

	FRM	RT	XGBoost	FRM-RT(%)	RT-XGB (%)
Mean squared error	0.073	0.068	0.042	-6	-39
Mean absolute error	0.205	0.180	0.127	-12	-29
Relative squared error (%)	0.483	0.452	0.276	-6	-39
Relative absolute error (%)	0.577	0.506	0.357	-12	-29
Correlation coefficient	0.721	0.745	0.852	3	14

TABLE 5.2: Comparison of predictive accuracy. Out-of-sample predictive accuracy measures of recovery rate predictions given by a fractional response model, a single regression tree and a XGBoost

A regression tree gives better predictions of recoveries than the fractional response model across all measures. For instance, the regression tree decreases the out-of-sample MSE of the complete data set by about 6% compared to the fractional response model. When comparing the XGBoost with the regression tree, we observe significant improvements in predictive accuracy. Across all measures, the XGBoost outperforms the regression tree model. There are reductions in MSE and MAE of about 39%, 29%, respectively. Accordingly, XGBoost presents lower relative errors. Finally, the recoveries predicted by the XGBoost also show a higher correlation with actual recoveries.

Figure 5.1 shows three histograms of the predicted recovery rate corresponding to the three different models, in relation to the actual recovery rate. Through Figure 5.1, it can

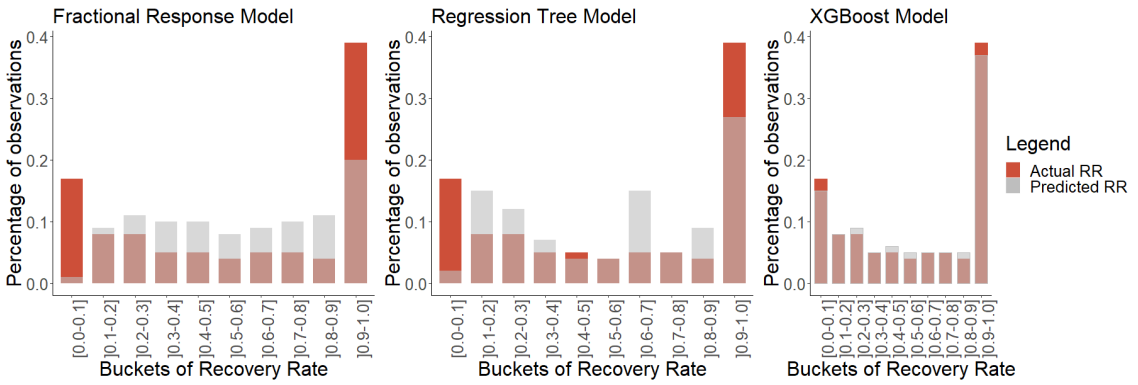

FIGURE 5.1: Predicted RR against actual RR on all data (n = 4630) using FRM, RT and XGB models

be concluded that the XGBoost model is better at predicting extreme recovery rate values (close to 0 or 1) than the Fractional Response Model or the Regression Tree. Overall, the XGBoost model appears to be more powerful in predicting both extreme and remaining values.

5.3 Global interpretability

This section aims to demonstrate how the selected interpretability methods add interpretability to the XGBoost model. In addition, it presents the interpretation process for natural interpretable models (FRM and RT) and compares it with that of XGBoost.

5.3.1 Natural Interpretable Models

Fractional Response Model

The marginal impact is calculated with the parameters of the regression reported in Table A.3, which reports the model coefficients that were obtained with the FRM when a logistic functional form is considered.

The partial effects results can be found in Table A.4. For a more visual interpretation of the results, Figure 5.2 presents the absolute average partial effects in decreasing order for the 15 variables with the greatest impact. Looking at Table 5.2, the effect of the obligor

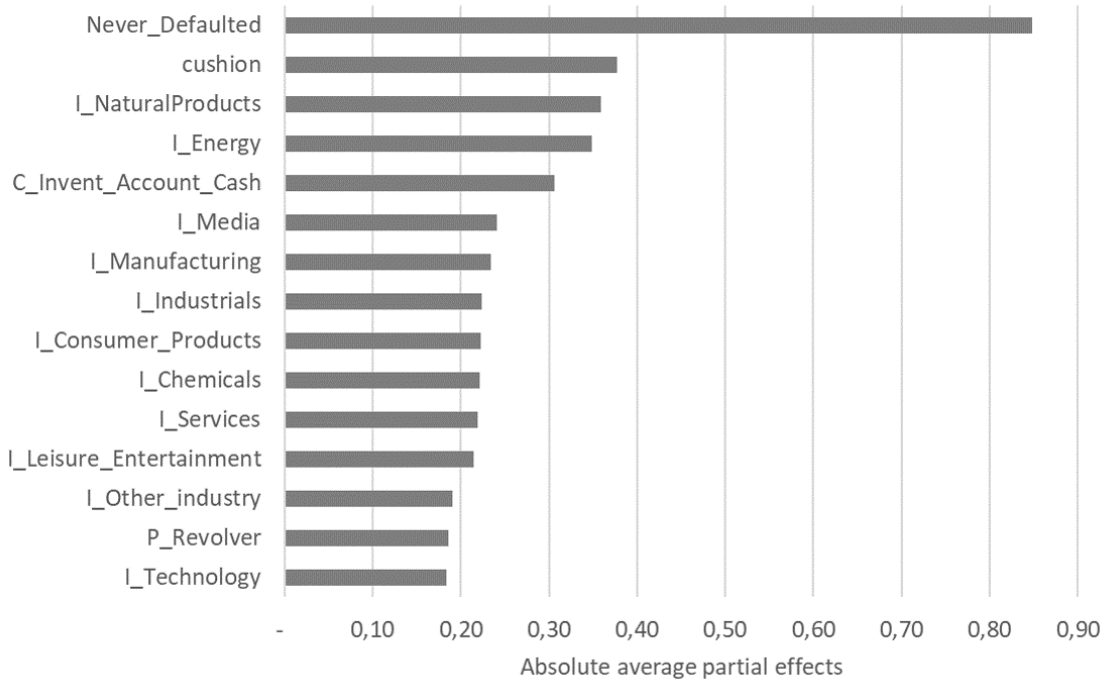

FIGURE 5.2: Top 15 variables with higher absolute average partial effects

having never defaulted previously ($never\_defaulted$) has the greatest significant impact on the recovery rate with an APE of 0.85 p.p.. The variable with the second highest impact is $cushion$, with a positive and significant APE of 0.38 p.p.. This means that the higher the percentage of junior outstanding debt to the defaulted instrument, the higher the expected recovery rate.

With regards to $I\_NaturalProducts$ one can conclude that a company in the Natural Products sector increases the recovery rate by 0.36 p.p. compared to a company in the Telecommunications sector. Most industries have larger recoveries than the reference group, with these differences being statistically significant. However, it can be observed that a company in the Environment sector has a negative and statistically significant impact, reducing the recovery rate by 0.05 p.p., on average, compared with a company in the Telecommunications sector.

It is also interesting to note that instruments guaranteed by inventories, accounts receivable, and cash have a higher positive effect (0.03 p.p.) than unsecured securities (the reference group). The remaining collateral flags are not significant or have minimal effect.

Additionally, the results indicate no significant difference in recoveries between the different types of subordinated bonds. The remaining types of instruments have significantly higher recoveries than the reference group (junior subordinated bonds).

The main advantage of using these types of models is that, looking directly at the APE table, it is quick and intuitive to conclude with respect to the variables that have impact on the model, as well as measuring the size and direction of impacts without much effort.

Regression Tree

The simplest way to assess the importance of variables in a regression tree model is by observing the variables that make up the tree and, in the case of more complex trees, by observing those at the top of the tree.

Starting from the same sample, the top of the tree will always be the same, and for clarity of illustration, a small and shallow tree was deliberately adjusted. This tree is illustrated in Figure 5.3 and may be interpreted in the following way. First, it is asked if the percentage of $cushion$ is less than 0.37 ($cushion$ < 0.37). If the answer is 'no', then the expected recovery rate is 0.86 (RR = 0.86), and the branch ends there. If the answer is 'yes' it is subsequently asked if the corporation has ever defaulted in the past ($Never\_defaulted$). If the answer is 'no', then the expected recovery rate is 0.99, and the branch ends there. If the answer is 'yes' and the percentage of $above$ is less than 0.075, the expected recovery rate is 0.58; if the answer is 'no', the percentage of $above$ is greater or equal to 0.075 and the corporation belongs to the energy sector, the expected recovery rate is 0.57. If, on the other hand, the corporation does not belong to the energy sector, then the expected recovery rate is 0.30.

Considerably higher recovery rates are expected in default events where: (i) the percentage of $cushion$ is higher than 0.37 (RR = 0.86) or (ii) the borrowing corporation has never been in default before (RR = 0.99). In these two situations, the predicted recovery rates are much higher than any other recovery rate predicted by the regression tree.

For default events where the percentage of $cushion$ is low and where the borrowing corporation has defaulted in the past, the recovery rate prediction is considerably lower compared to the remaining possibilities, if: (i) the percentage of debt senior is greater than 0.075 and (ii) the corporation holding the loan is not in the energy sector.

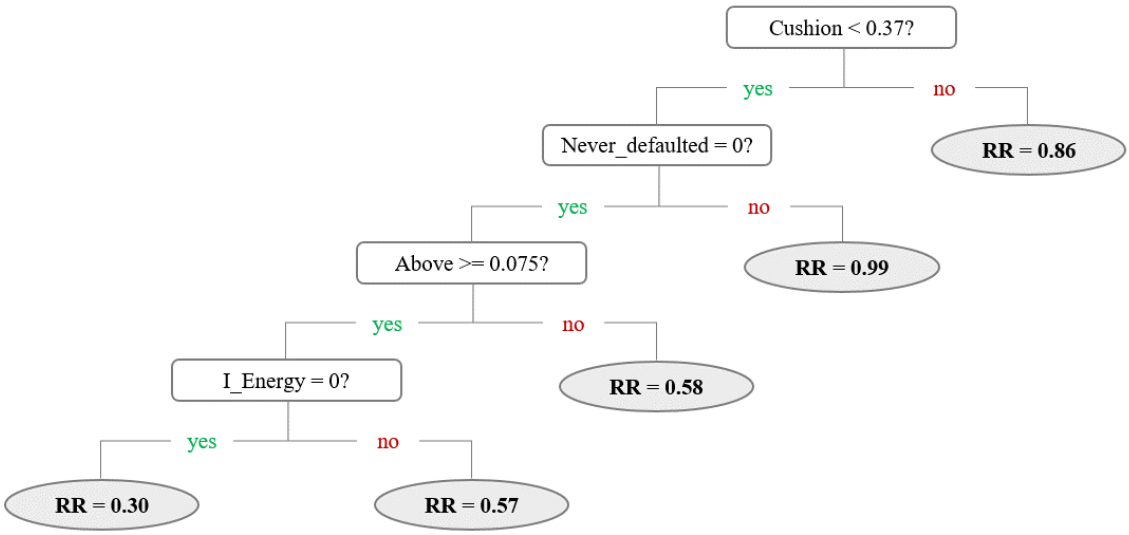

FIGURE 5.3: Regression tree to predict recovery rates from Moody's Ultimate Recovery Database. This tree was estimated with the 4630 observations in the database

One of the properties of the regression tree model is that it selects the most relevant variables for the prediction. In the case of Figure 5.3, the $cushion$, $Never\_defaulted$, $Above$ and I_Energy variables have been selected by the regression tree as the most relevant variables for the prediction of RR.

Since the full tree generated by the model is large and takes into account the contribution of all variables along the tree, a metric that considers the reduction of the sum of squares in each division was used to analyze the most important variables to the model. This metric allows for the accounting of the importance of all the variables in the tree, unlike simple looking at the top. This is important because, in some cases, a variable can be used in several divisions of the tree, and the sum of its contributions over its divisions can be greater than the contribution of a variable that is used only once with a greater reduction when compared separately with the remaining splits. Metric values are standardized against the most important regressor (so that it has a value of 100) and the remaining regressors are scored based on their relative reduction in the loss function. Figure 5.4 shows the results obtained. Only variables with more than 1% of relative importance appear represented with a bar in the graph. It is concluded that, except for I_Energy, all variables identified as most important through the analysis of the top of the tree (Figure 5.3) remain important when considering this metric. Additionally, there are variables that, despite not

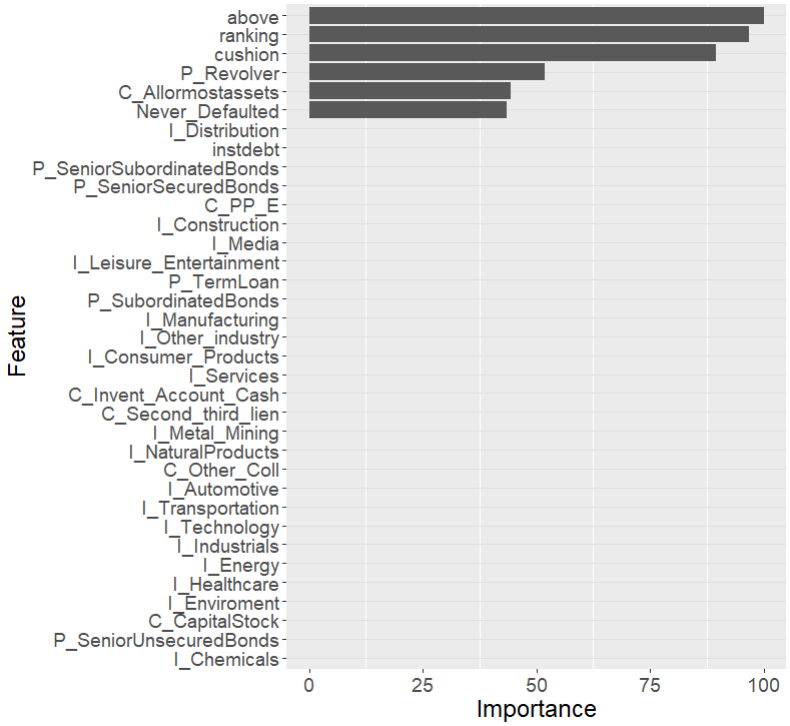

FIGURE 5.4: Importance of regressors based on the sum of squares reduction in splits

being selected in the tree's initial splits, provide valuable contributions to the tree, as is the case with $C\_Allormostassets$ and $P\_Revolver$.

Thus, through visual analysis of the tree and the construction of simple metrics, it is easy to identify the most impactful variables and the direction of their effects on the estimation of the regression tree model.

5.3.2 Black-box Model

XGBoost - eXtreme Gradient Boosting

For a first understanding of XGBoost at the global level, the classical feature importance metrics generally applied to the models based on decision trees are considered, namely: Frequency, Gain and Cover. Figure 5.5 displays the results obtained for these metrics.

Through the analysis of Figure 5.5, it can be observed that $instdebt$ is used more frequently than all of the other variables (Frequency) while also impacting a significant proportion of the observations (Cover). However, $instdebt$ ranks third on the gain metric. This may be because this variable has been used many times, but at a deeper level in the tree, after the sample has been previously divided by other variables.

The $never\_defaulted$ variable was not often used as a splitting variable (ranked penultimate in the Top 15 by Frequency). However, this variable leads to a significant

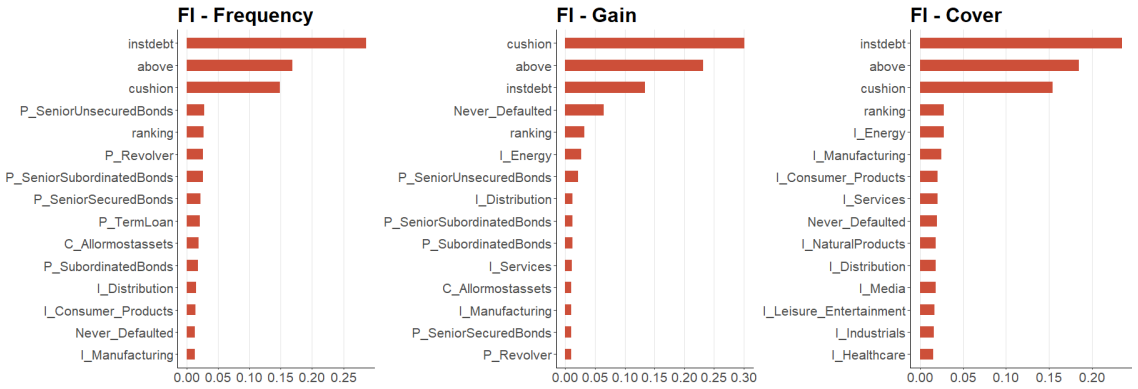

FIGURE 5.5: Classical Feature Importance metrics

increase in terms of gain, even though it was used a reduced number of times compared to the other variables.

It is interesting to note that the Top 3 of all metrics include the variables $instdebt$, $above$ and $cushion$. Through the analysis of classical measurements, these three variables are notoriously the most important for estimating recovery rates. Nonetheless, it is important to note that these metrics are not always consensual and may lead to very different rankings between the different measures. Additionally, these metrics do not produce results that allow for the comparison between models and lead to an incorrect assessment of the most relevant variables. For example, some splits with certain regressors (which these measures consider to be non-significant) may be contributing to an information gain along the path created by the division of the variable. These metrics do not take this type of interactions into account.

The results obtained by applying the permutation feature importance method are displayed in Figure 5.6. The variable identified as the most important with this method is the $cushion$ variable, since it lead to an increase in MAE by a factor of 8.60. The second and third most important variables are the $above$ and $instdebt$ variables, generating an increase in the MAE by a factor of 7.14 and 5.36, respectively. On the other hand, the less important variable is C_Other_Coll, which only produces an increase in MAE by a factor of 1.06.

Although the permutation technique takes into account the model's interactions based on the decrease in model performance, it does not identify the importance of the variables through how much the model's output varies for a regressor. For this, the SHAP value was applied, since it is based on the magnitude of regressors contributions to prediction.

Figure 5.7 presents the results obtained for the SHAP absolute importance. The two variables identified as the most relevant are the $cushion$ and $above$ variables. It is interesting to note that these are also identified as the most important variables in both the Gain metric and in the Permutation Feature Importance method.

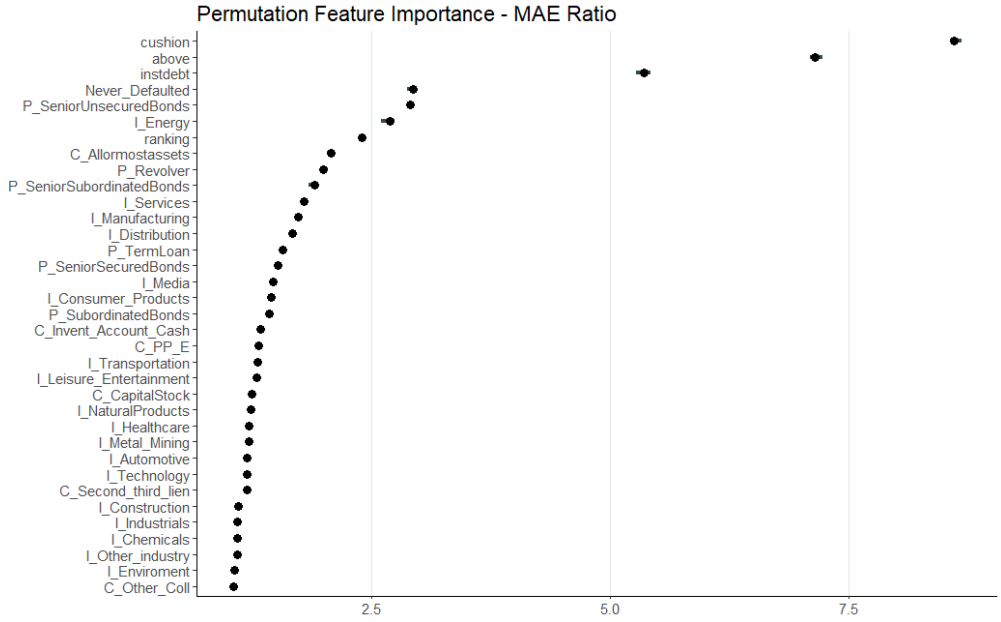

FIGURE 5.6: Permutation feature importance - MAE Ratio

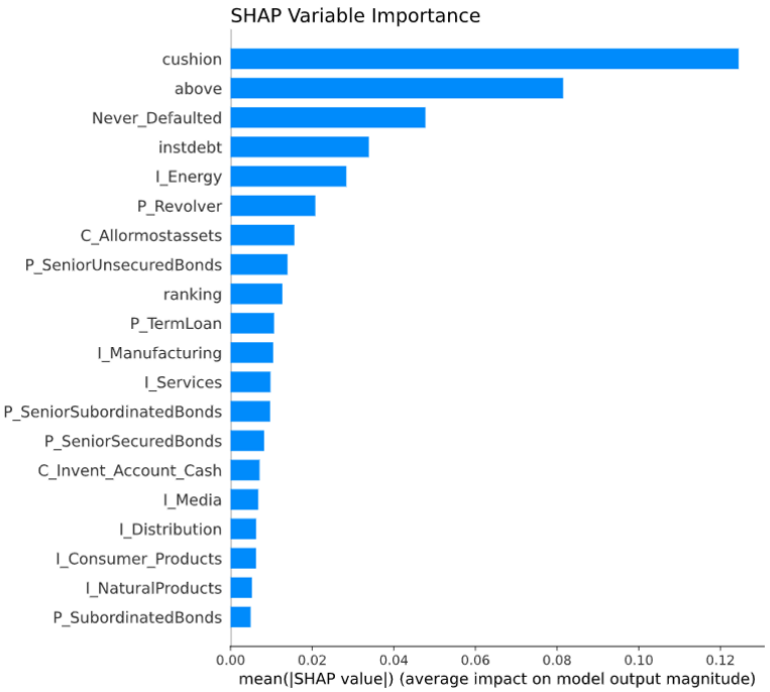

FIGURE 5.7: SHAP Feature Importance

The $instdebt$ variable, which was identified as the most relevant variable in both Frequency and Cover metrics, falls to fourth place in the SHAP value metric, reinforcing the idea that both the frequency of use of variables in splits (Frequency) and the number of observation included in a regressors' splits (Cover) do not impact the final prediction

directly.

The analyses presented above are useful, but can only assess with regards to feature importance. To analyze the marginal effects of the most important variables on the predicted outcome of the XGBoost model, Figure 5.8 presents the partial dependence plots (PDP) for the five variables with the highest average absolute SHAP value.

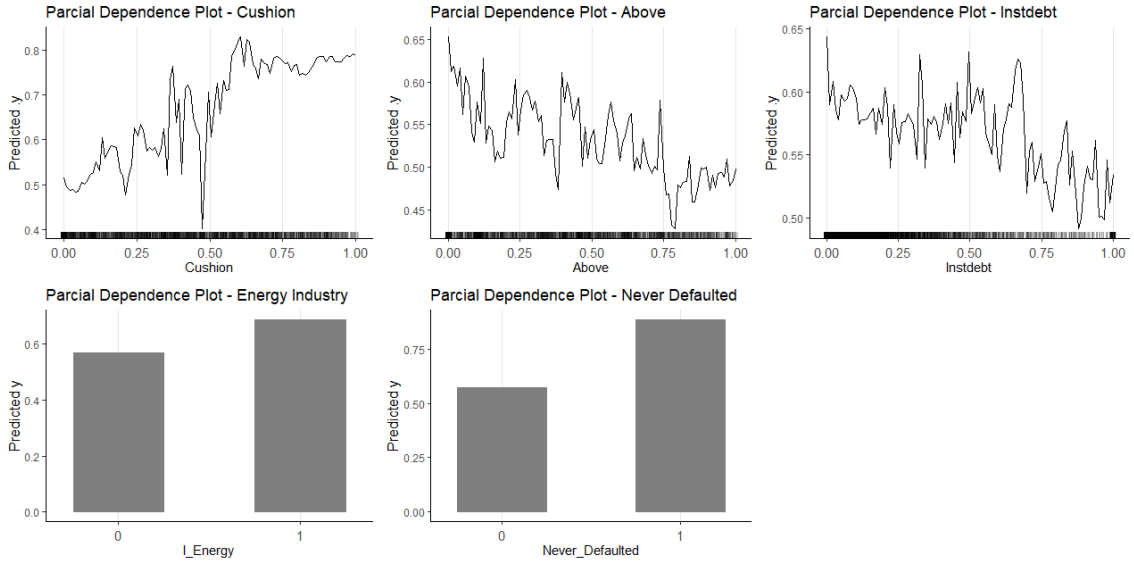

FIGURE 5.8: Partial Dependence Plots

Through the analysis of the PDP, it can be observed that, on average, the model predicts higher recoveries when: (i) the percentage of below ($cushion$) is high; (ii) the percentage of $above$ is low; (iii) the outstanding amount at default relative weight in obligor's total ($instdebt$) is low; (iv) the obligor has never been in default before ($never\_defaulted$) and (v) the defaulted instrument is from the Energy sector (I_Energy).

Observing the scale of the respective average recovery rate variation for all variables, it can be concluded that $cushion$ has the most significant range, followed by $above$ and $Never\_defaulted$, presenting a pattern equal to the classification of the SHAP method previously presented.

5.3.3 Global interpretability between models

Table 5.3 presents a summary of the top 5 most important variables selected by each model. The analysis of this table allows us to observe that, for the data under analysis, the 3 most relevant variables are quite consensual throughout the models and feature importance methods applied in this study. This can be seen since the variables $cushion$, $above$ and $never\_defaulted$ always end up on the top 5, while at least one of them also ends up being classified as the most important variable by the various models and metrics.

	FRM - Abs(APE)	RT - SSR	XGBoost - PFI	XGBoost - SHAP
Cushion	2^{nd}	3^{rd}	1^{st}	1^{st}
Above		1^{st}	2^{nd}	2^{nd}
never_defaulted	1^{st}		4^{th}	3^{rd}
instdebt			3^{rd}	4^{th}
Ranking		2^{nd}		
I_energy	4^{th}			5^{th}
P_Revolver		4^{th}		
P_SeniorUnsecuredBonds			5^{th}	
C_Allmostassets		5^{th}		
I_NaturalProducts	3^{rd}			
C_Invent_Account_Cash	5^{th}			

TABLE 5.3: Comparison of the most important variables selected by each models

An interesting observation is that the outstanding amount at default relative weight in obligor's total debt ($instdebet$) appears in the top 5 most important variables for the Black-box model but apparently it is not relevant in the natural interpretable models (presenting a very small partial effect in the FRM and not being part of the variables considered as most important by the RT).

Considering the metrics used for the global interpretation of the models, it is clear that the natural interpretable models allow a faster and more measurable understanding of the impact of each regressor on the models. This type of interpretation does not involve extra effort in obtaining additional metrics to those provided by the models.

On the other hand, it is observed that the global analysis of a black-box model (in this case, the XGBoost) requires an understanding of more complex metrics and an additional effort in their calculation. The diversity of metrics available to assess the importance of the black-box models' variables also bring some uncertainty about the most suitable method. However, if the analysis objective is well defined, it is easier to choose the metric to apply. For example, if the objective is to measure the variables' contribution to the prediction, SHAP is a good metric to consider. If the objective is to measure the variables' contribution in reducing the error, PFI is a good choice.

With regards to the sign of the effects through the observation of the results obtained in the PDP and the average partial effects of the FRM, it is concluded that the sign of the effects of the most important variables identified by the SHAP methodology is consistent between the two models and is in accordance to economic theory (see Table 5.4).

Regressor	XGBoost - PDP	FRM - APE
Cushion	(+)	(+)
Above	(-)	(-)
Instdebt	(-)	(-)
Never_defaulted	(+)	(+)
I_Energy	(+)	(+)

TABLE 5.4: Sign of the effects of the most important variables identified by the SHAP methodology for the XGBoost model and those obtained in the FRM

6. CONCLUSION

This work compares the use of natural interpretable models (Fractional Response Model and Regression Tree Model) with a Black-Box model (Extreme Gradient Boosting) in relation to the quality of the predictions and its interpretation applied to the Loss Given Default. This is a topic of great interest to financial institutions, since it has a direct impact on the calculation of the minimum regulatory capital requirements. More efficient estimates combined with higher levels of interpretability of the factors inherent to higher credit losses pave the way for the achievement of comparative advantages by the financial institution in the Financial Market.

Using data from Moody's Ultimate Recovery Database, the present work has shown that a well-tuned XGBoost model outperforms the natural interpretable models in all the metrics considered. Additionally, it is concluded that XGBoost is better at predicting both the extreme values (resulting from the bimodal distribution of the recovery rate) and the non-extreme values.

With regards to interpretability, the present work sought to understand the internal workings of each model at a global level. For the Fractional Response Model, this was accomplished throught the use of average partial effects (APE), while for the Regression Tree it was done by observing the tree and assessing the decrease in the sum of squares in each split across the entire tree. Lastly, for XGBoost, the Permutation Feature Importance method, the SHAP method, and the construction of Partial Dependence Plots were used.

For the natural interpretable models, it was possible to extract quantifiable rules only through the analysis of the partial effects of the FRM model, and for the RT model, the quantifiable rules were extracted through the analysis of the tree. For these models, it is easy to present the extracted rules so that they are intuitive from a human point of view without much additional effort. On the other hand, it can be observed that the global analysis of XGBoost requires an understanding of more complex metrics and an additional

effort in their calculation. However, after these learning costs have been overcomed, the predictions provided by the black-box models for Loss Given Default can be easily interpreted in terms of their inputs.

REFERENCES

Acharya, V., Bharath, S. and Srinivasan, S. (2007), 'Does industry-wide distress affect default loans? evidence from creditor recoveries.', *Journal of Financial Economics, 85, 787-821* .

Altman, E. I. and Kalotay, E. A. (2014), 'Ultimate recovery mixtures', *Journal of Banking and Finance 40, 116-120* .

Altman, E. and Kishore, V. (1996), 'Almost everything you wanted to know about recoveries on defaulted bonds', *Financial Analysts Journal, 52* .

Asarnow, E. and Edwards, D. (1995), 'Measuring loss on defaulted bank loans: A 24 year study', *Journal of Commercial Lending 77(7), 11-23* .

Bastos, J. A. (2010), 'Forcasting bank loans loss-given-default', *Journal of Banking & Finance 34, 2510-2517* .

Bastos, J. A. (2014), 'Ensemble predictions of recovery rates', *Journal of Financial Services Research 46, 177-193* .

BCBS (1988), 'International convergence of capital measurement and capital standards', `https://www.bis.org/publ/bcbs04a.htm` .

BCBS (2004), 'International convergence of capital measurement and capital standards', `https://www.bis.org/publ/bcbsca.htm` .

Bracke, P., Datta, A., Jung, C. and Sen, S. (2019), 'Machine learning explainability in finance: an application to default risk analysis', *Bank of England, Staff Working Paper No. 816* .

Breiman, L. (2001), 'Random forests', *Machine Learning 45 (1), 5-32* .

Breiman, L., Friedman, J. H.and Olshen, R. A. and Stone, C. J. (1984), *Classification and regression trees*, Monterey, CA: Wadsworth and Brooks.

Caselli, S., Gatti, S. and Querci, F. (2008), 'The sensitivity of the loss given default rate to systematic risk: New empirical evidence on bank loans', *Journal of Financial Services Research 34, 1-34* .

Chen, T. and Guestrin, C. (2016), 'Xgboost: A scalable tree boosting system'.

Craven, M. and Shavlik, J. W. (1996), 'Extracting tree-structured representations of trained networks', *Advances in neural information processing systems, 24-30* .

Davydenko, S. and Franks, J. (2008), 'Do bankruptcy codes matter? a study of defaults in france, germany, and the u.k.', *The Journal of Finance 63, 565-608* .

Dennis, J. and Schnabel, R. (1996), 'Convergence results for properly chosen steps.', *In Numerical methods for unconstrained optimization and nonlinear equations, 120-125* .

Dermine, J. and Neto de Carvalho, C. (2006), 'Bank loan losses-given-default: A case study', *Journal of Banking and Finance 30, 1219-1243* .

Fisher, A., Rubin, C. and Dominici, F. (2018), 'Model class reliance: Variable importance measures for any machine learning model class, from the 'rashomon' perspective'.

Franks, J., de Servigny, A. and Davydenko, S. (2004), 'A comparative analysis of the recovery process and recovery rates for private companies in uk, france and germany', *Standard and Poor's Risk Solutions.* .

Friedman, J. H. (2001), 'Greedy function approximation: A gradient boosting machine.', *The Annals of Statistics, 29(5), 1189-1232* .

Greenwell, B., Boehmke, B. and McCarthy, A. (2001), 'A simple and effective model-based variable importance measure', *arXiv preprint arXiv:1805.04755* .

Grossman, R., O'Shea, S. and Bonelli, S. (2001), 'Bank loan and bond recovery study: 1997-2000', *Fitch Loan Products Special Report, March* .

Grunert, J. and Weber, M. (2009), 'Recovery rates of commercial lending: Empirical evidence for german companies', *Journal of Banking & Finance 33, 505-513* .

Guidotti, R., Monreale, A., Ruggieri, S., Turini, F., Giannotti, F. and Pedreschi, D. (2018), 'A survey of methods for explaining black box models', *ACM Computing Surveys 51(5), 1–42* .

Gupton, G., Gates, D. and Carty, L. (2000), 'Bank loss given default.', *Global Credit Research, Moody's* .

Gupton, G. M. and Stein, R. M. (2005), 'Losscalc v2: Dynamic prediction of lgd', *Moody's KMV* .

Kearns, M. (1988), 'Thoughts on hypothesis boosting (tech. rep.)', *University of Pennsylvania* .

Kearns, M. and Valiant, L. G. (1989), 'Crytographic limitations on learning boolean formulae and finite automata'.

Kim, J. (2009), 'Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap', *Computational Statistics & Data Analysis, 53(11), 3735-3745* .

Loterman, G., Brown, I., Martens, D., Mues, C. and Baesens, B. (2012), 'Benchmarking regression algorithms for loss given default modeling', *International Journal of Forecasting 28, 161-170* .

Lundberg, S. and Lee, S. (2017), 'A unified approach to interpreting model predictions', *Advances in neural information processing systems* .

Miller, T. (2019), 'Explanation in artificial intelligence: Insights from the social sciences', *Artificial Intelligence, 267, 1–38* .

Molinaro, A. (2005), 'Prediction error estimation: A comparison of resampling methods', *Bioinformatics, 21(15), 3301–3307* .

Molnar, C. (2019), *Interpretable Machine Learning*, github. `https://christophm.github.io/interpretable-ml-book/`.

Papke, L. E. and Wooldridge, J. M. (1996), 'Econometric methods for fractional response variables with an application to 401(k) plan participation rates', *Journal of Applied Econometrics 11, 619–632* .

Qi, M. and Zhao, X. (2011), 'Comparison of modeling methods for loss given default', *Journal of Banking & Finance 35, 2842-2855* .

Ribeiro, M., Singh, S. and Guestrin, C. (2016), '"why should i trust you?": Explaining the predictions of any classifier'.

Schapire, R. E. (1990), 'The strength of weak learnability', *Machine Learning, 5, 197-227* .

Shapley, L. (1953), 'Contributions to the theory of games', *Kuhn, H. and Tucker, A., Eds., Contributions to the Theory of Games II, Princeton University Press, Princeton, 307-317* .

Sigrist, F. (2018), 'Gradient and newton boosting for classification and regression'.

Varma, P. and Cantor, R. (2005), 'Determinants of recovery rates on defaulted bonds and loans for north american corporate issuers: 1983–2003', *J Fixed Income 14, 29–44* .

Yang, C., Rangarajan, A. and Ranka, S. (2018), 'Global model interpretation via recursive partitioning'.

Yao, X., Crook, J. and Andreeva, G. (2015), 'Support vector regression for loss given default modelling', *European Journal of Operational Research 240(2), 528-538* .

APPENDIX

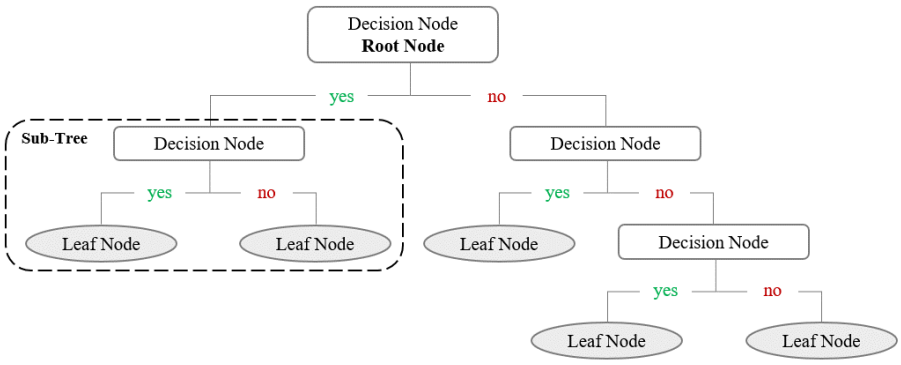

FIGURE A.1: Usual structure of a regression tree

Hyperparameter	Definition	Proposed setting
Learning rate	Shrinks the weights predicted by each tree. This is a fixed value in order to avoid overfitting and thus make the model more conservative.	0.01, 0.05, 0.1, and 0.2
Minimum child weight	Gives the minimum number of instances required in a child node. If this value increases, it makes the model more conservative.	1,2,4 and 6
Maximum tree depth	Indicates the maximum amount of edges between the root node of a regression tree and its nodes. Increasing this value will make the model more complex and more likely to overfit.	4 to 14 with step size 2
Column sample percentage	Subsample ratio of columns when constructing each tree. Subsampling will occur once in every boosting iteration.	0.5, 0.9 and 1
Sub sample percentage	Subsample ratio of the training instances. Setting it to 0.5 means that XGBoost would randomly sample half of the training data prior to growing trees and this will prevent overfitting. Subsampling will occur once in every boosting iteration.	0.5, 0.9 and 1
Gamma	States the minimum loss reduction needed in order for a tree to make a new split. If the value for gamma is higher, trees become more shallow.	0 to 8, with step size 2

TABLE A.1: XGBoost Hyperparameters

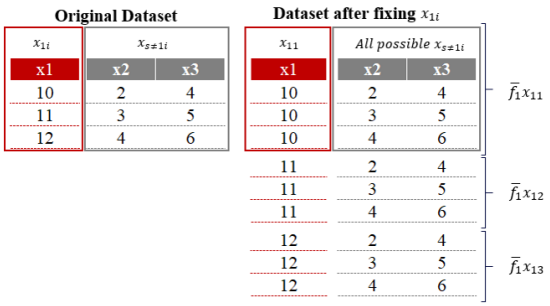

FIGURE A.2: Simplified representation of PDP calculation

Input:
trained model f
feature matrix X
target vector y
error measure $L(y, f)$
Process:
for each feature $j = 1, ..., p$
generate feature matrix X^{perm} by permuting feature j in the data X (this breaks the association between feature j and true outcome y)
estimate error $e^{perm} = L(Y, f(X^{perm}))$ based on the predictions of the permuted data
Output:
calculate permutation feature importance
$FI^j = e^{perm}/e^{orig}$ or
$FI^j = e^{perm} - e^{orig}$
sort features by descending FI

TABLE A.2: Pseudocode for the permutation feature importance algorithm based on Fisher et al. (2018)

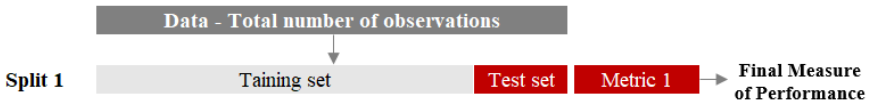

FIGURE A.3: Holdout method scheme

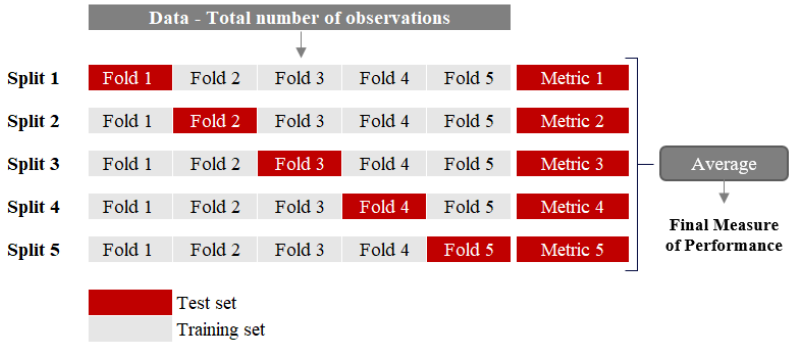

FIGURE A.4: K-fold cross-validation method schema

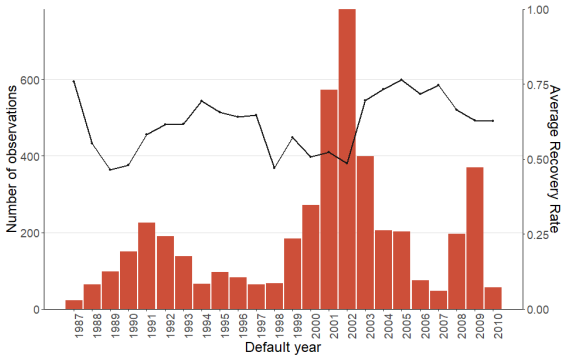

FIGURE A.5: Number of defaulted instruments (Bars) and average recovery rate (Line) by year of default in Moody's Ultimate Recovery Database (1987-2010)

Variable	Estimate	Variable	Estimate
INTERCEPT	-1.975 (0.000***)	I_Technology	1.135 (0.000***)
I_Transportation	1.002 (0.000***)	P_Revolver	1.148 (0.000***)
I_Automotive	1.075 (0.000***)	P_SeniorSecuredBonds	0.667 (0.016**)
I_Chemicals	1.367 (0.000***)	P_SeniorSubordinatedBonds	0.100 (0.692)
I_Construction	0.99 (0.000***)	P_SeniorUnsecuredBonds	0.837 (0.001***)
I_Consumer_Products	1.374 (0.000***)	P_SubordinatedBonds	0.373 (0.143)
I_Distribution	0.852 (0.000***)	P_TermLoan	0.821 (0.002***)
I_Energy	2.153 (0.000***)	C_Allormostassets	0.466 (0.001***)
I_Environment	-0.325 (0.097*)	C_CapitalStock	0.194 (0.256)
I_Healthcare	0.908 (0.000***)	C_Invent_Account_Cash	1.894 (0.000***)
I_Industrials	1.384 (0.000***)	C_Other_Coll	0.497 (0.094*)
I_Leisure_Entertainment	1.324 (0.000***)	C_PP_E	0.166 (0.301)
I_Manufacturing	1.45 (0.000***)	C_Second_third_lien	0.095 (0.556)
I_Media	1.488 (0.000***)	above	-0.848 (0.000***)
I_Metal_Mining	1.092 (0.000***)	cushion	2.334 (0.000***)
I_NaturalProducts	2.223 (0.000***)	ranking	-0.087 (0.034**)
I_Other_industry	1.176 (0.000***)	instdebt	-0.217 (0.040**)
I_Services	1.356 (0.000***)	Never_Defaulted	5.246 (0.000***)
RESET	4.588 (0.101)		

TABLE A.3: Model coefficients given by a fractional response regression. The p-values are shown in parenthesis. A logistic functional form was used. legend: * p<.1; ** p<.05; *** p<.01

Variable	Estimate	Variable	Estimate
I_Technology	0.183 (0.000***)	I_Transportation	0.162 (0.000***)
I_Automotive	0.174 (0.000***)	P_Revolver	0.186 (0.000***)
I_Chemicals	0.221 (0.000***)	P_SeniorSecuredBonds	0.108 (0.015**)
I_Construction	0.160 (0.000***)	P_SeniorSubordinatedBonds	0.016 (0.692)
I_Consumer_Products	0.222 (0.000***)	P_SeniorUnsecuredBonds	0.135 (0.001***)
I_Distribution	0.138 (0.000***)	P_SubordinatedBonds	0.060 (0.143)
I_Energy	0.348 (0.000***)	P_TermLoan	0.133 (0.002***)
I_Environment	-0.053 (0.097*)	C_Allormostassets	0.075 (0.001***)
I_Healthcare	0.147 (0.000***)	C_CapitalStock	0.031 (0.256)
I_Industrials	0.224 (0.000***)	C_Invent_Account_Cash	0.306 (0.000***)
I_Leisure_Entertainment	0.214 (0.000***)	C_Other_Coll	0.080 (0.095*)
I_Manufacturing	0.234 (0.000***)	C_PP_E	0.027 (0.302)
I_Media	0.241 (0.000***)	C_Second_third_lien	0.015 (0.556)
I_Metal_Mining	0.177 (0.000***)	above	-0.137 (0.000***)
I_NaturalProducts	0.359 (0.000***)	cushion	0.377 (0.000***)
I_Other_industry	0.190 (0.000***)	ranking	-0.014 (0.034**)
I_Services	0.219 (0.000***)	Never_Defaulted	0.848 (0.000***)
instdebt	-0.035 (0.040**)		

TABLE A.4: Model Average Partial Effects (APE) given by a fractional response regression. The p-values are shown in parenthesis. legend: * p<.1; ** p<.05; *** p<.01